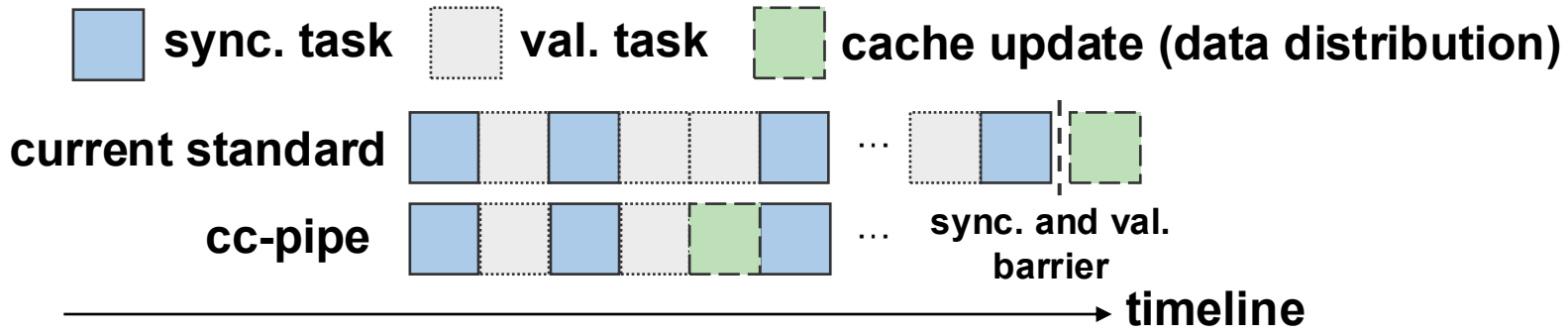
IETF Hackathon

cc-pipe: Concurrent and Conflict-free Pipeline for RPKI Relying Parties

IETF 125
14–15 March 2026
Shenzhen, China

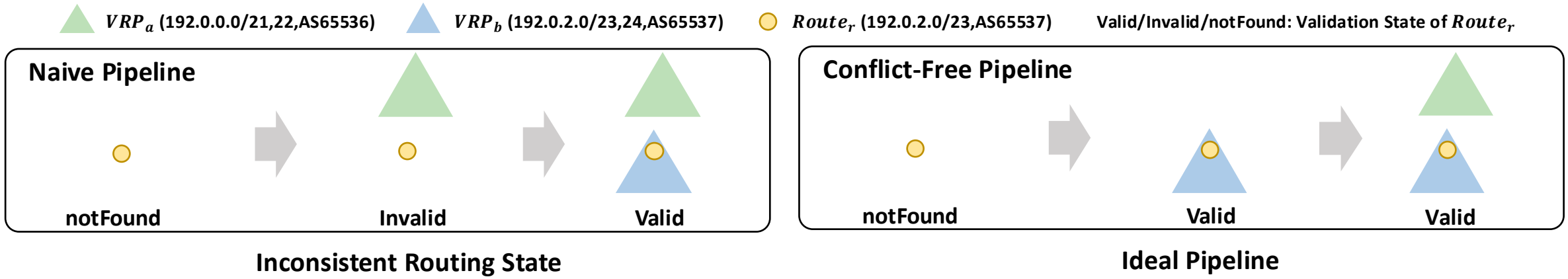
Hackathon Plan

- **What problem are we working on?**
  - Systemic barriers in the RPKI data supply chain. Traditional RPs block all updates from being distributed to routers until all data is synchronized and validated, delaying critical RPKI data updates.
- **Related RFC**
  - RFC8210



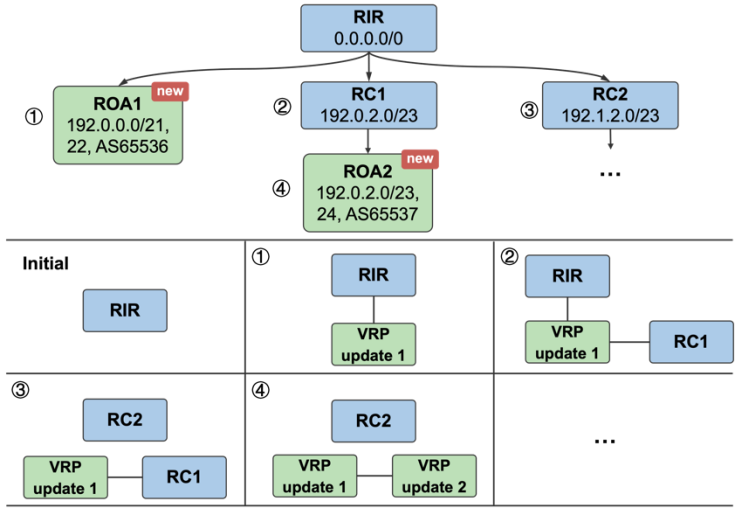
Hackathon Plan

- **Our Goal**
  - Pipeline the RP validation run while preserving routing state consistency and mitigating introduced router overhead.
- **Our Approach**
  - **For consistency:** utilize the resource allocation rule (resource of a cert is a subset of its issuer) to prevent data conflict.
  - **For router overhead:** aggregate some updates and extend VRP cache maintenance strategy.

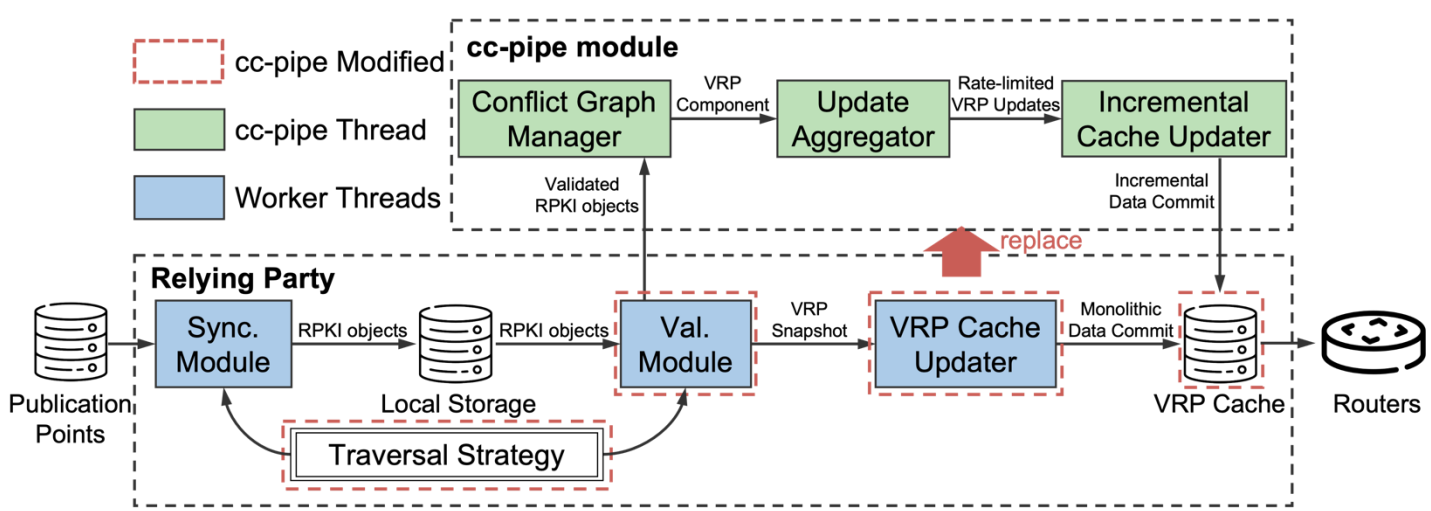


What got done

- **Key Results**
  - **Our design**
    - Conflict Graph to preserve consistency.
    - Strategies to mitigate router overhead.
    - cc-pipe architecture.



conflict graph example



cc-pipe architecture

What got done

- **Key Results**
  - **Evaluation**
    - Up to 73.3% RP validation run latency reduction and superior scalability and robustness.
    - Negligible router overhead in two open-source routers (FRRouting and BIRD).

What we learned

- **Lessons learned**
  - To explore the standardization of cc-pipe, we need more interoperability testing with a broader range of BGP router implementations (both open-source and commercial) to validate edge-case behaviors.

Wrap Up

Team members:

- Chenhui Yu (chyu@cnic.cn)
- Hui Zou (zouhui@cnic.cn)
- Zhuoran Ma (mazhuoran@hnu.edu.cn)

First timers @ IETF/Hackathon:

- All of us.

We have implemented cc-pipe based on two open-source RPs (Routinator and FORT-Validator), and our code is available at

https://github.com/FIRLab-CNIC/cc-pipe